USEFUL FOR UNIVERSITY EXAMS, GATE, NET AND OTHER CS EXAMS

DATABASE MANAGEMENT SYSTEM

Entity-Relationship Model with EXAMPLE

PROPER NOTES IN PPT FORM

PART -9



ER Model

- Entity Relational model is a model for identifying entities to be represented in the database and representation of how those entities are related.
- Creating an ER Model in DBMS is considered a best practice before implementing your database because it makes it easier for the developers to understand the database system just by looking at the ER model.
- ER model makes use of ER diagrams, which are the diagrams sketched to design a database.
- The ER data model specifies enterprise schema that represents the overall logical structure of a database graphically.
- E-R diagrams are used to model real-world objects like a person, a car, a company and the relation between these real-world objects.
- Entity Relational model is a model for identifying entities to be represented in the database and representation of how those entities are related.
- The ER data model specifies schema that represents the overall logical structure of a database graphically.
- E-R diagrams are used to model real-world objects like a person, a car, a company and the relation between these real-world objects.

Features of ER model

- E-R diagrams are used to represent E-R model in a database, which makes them easy to be converted into relations (tables).
- E-R diagrams provide the purpose of real-world modeling of objects which makes them useful.
- E-R diagrams require no technical knowledge and no hardware support.
- These diagrams are very easy to understand and easy to create even by a naive user.
- It gives a standard solution of visualizing the data logically.

Components of ER Model

Components of E-R Model

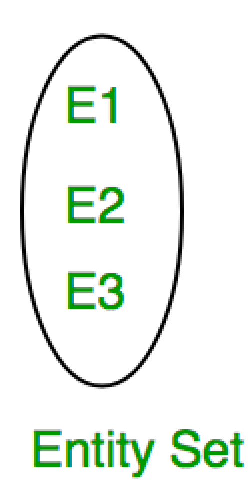
1-Rectangles	Represents entity
2-Ellipse	Represents attributes
3-Diamonds	Represents relationship among entities
4-Lines	link attributes to entities and entity sets to relationship

Entity, Entity Type, Entity Set –

- An Entity may be an object with a physical existence – a particular person, car, house, or employee – or it may be an object with a conceptual existence – a company, a job, or a university course.
- An Entity is an object of Entity Type and a set of all entities is called as an entity set. e.g.; E1 is an entity having Entity Type Student and set of all students is called Entity Set. In ER diagram, Entity Type is represented as:

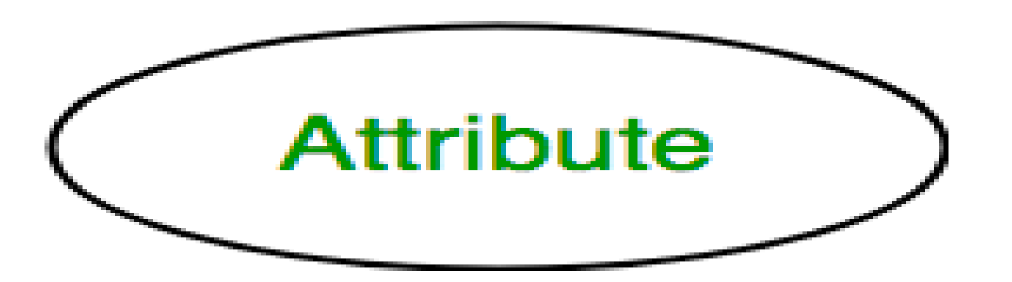
Student

Entity Type



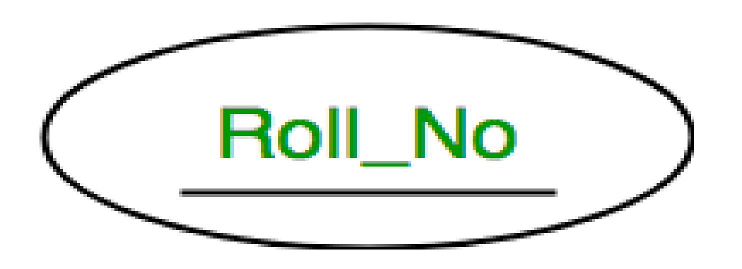
Attribute(s):

- Attributes are the **properties that define the entity type**. For example, Roll_No, Name, DOB, Age, Address, Mobile_No are the attributes that define entity type Student.



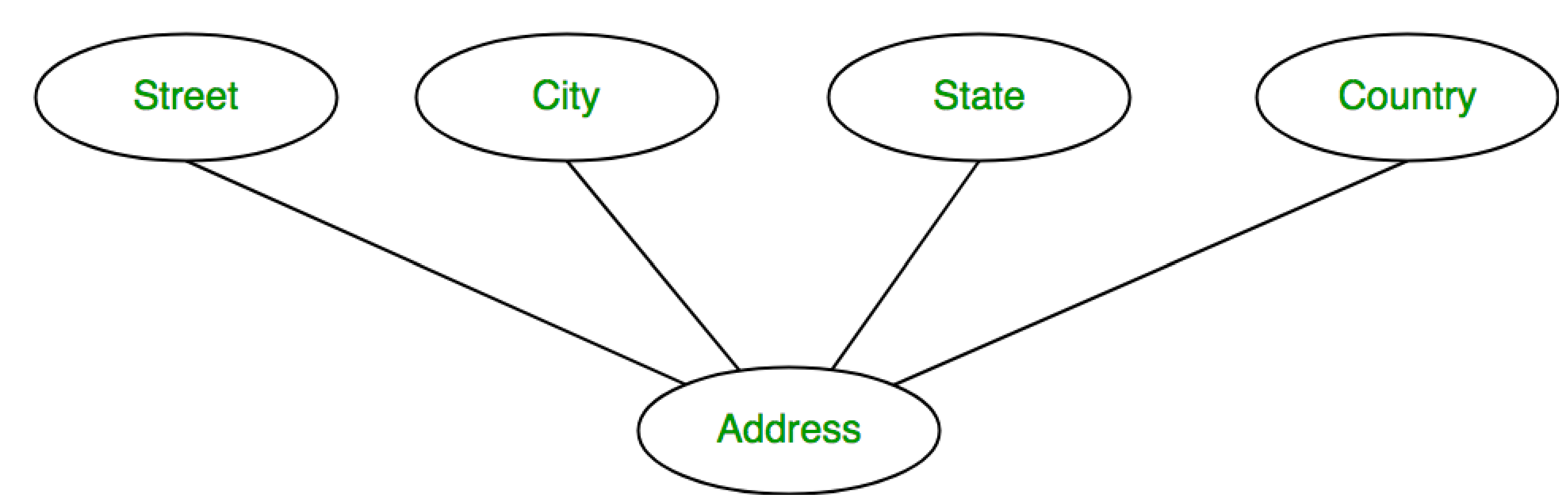
1. Key Attribute –

- The attribute which **uniquely identifies each entity** in the entity set is called key attribute.For example, Roll_No will be unique for each student. In ER diagram, key attribute is represented by an oval with underlying lines.



2. Composite Attribute –

- An attribute **composed of many other attribute** is called as composite attribute. For example, Address attribute of student Entity type consists of Street, City, State, and Country. In ER diagram, composite attribute is represented by an oval comprising of ovals.



3. Multivalued Attribute

- An attribute consisting **more than one value** for a given entity. For example, Phone_No (can be more than one for a given student). In ER diagram, a multivalued attribute is represented by a double oval.



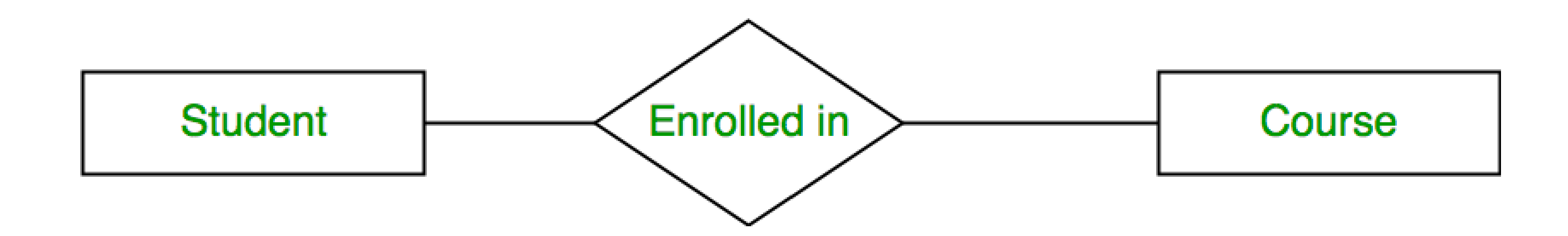
4. Derived Attribute –

- An attribute that can be **derived from other attributes** of the entity type is known as a derived attribute. e.g.; Age (can be derived from DOB). In ER diagram, the derived attribute is represented by a dashed oval.



Relationship Type and Relationship Set:

- A relationship type represents the **association between entity types**. For example,'Enrolled in' is a relationship type that exists between entity type Student and Course. In ER diagram, the relationship type is represented by a diamond and connecting the entities with lines.

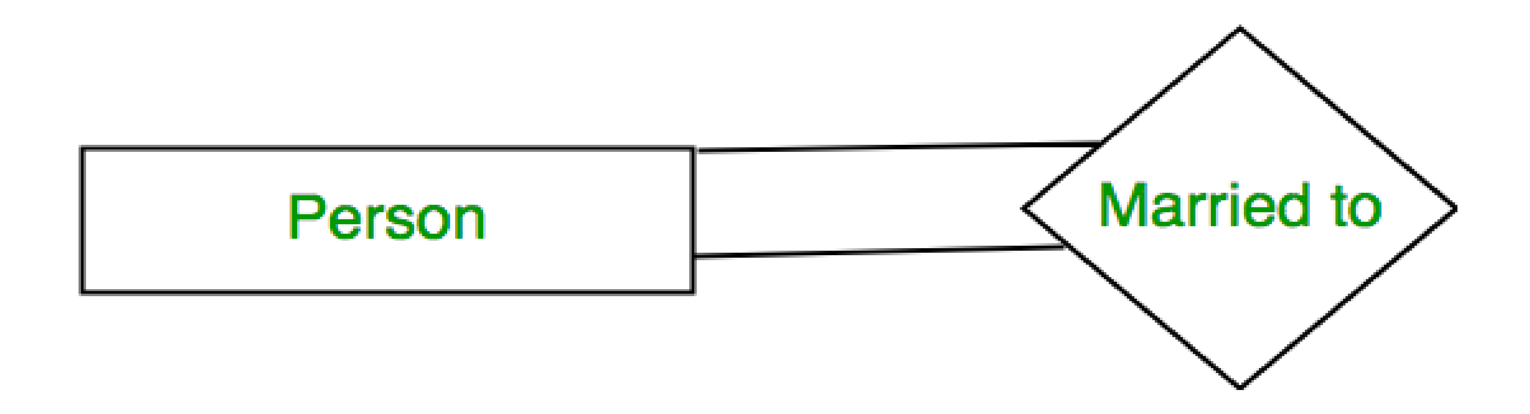


Degree of a relationship set:

- The number of different entity sets **participating in a relationship** set is called as the degree of a relationship set.

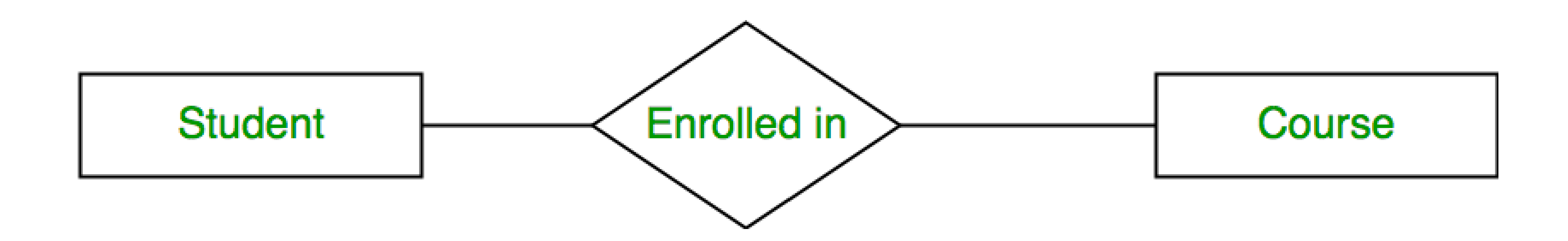
1. Unary Relationship –

- When there is **only ONE entity set participating in a relation**, the relationship is called a unary relationship. For example, one person is married to only one person.



2. Binary Relationship –

- When there are **TWO entities set participating in a relationship**, the relationship is called a binary relationship. For example, a Student is enrolled in a Course.

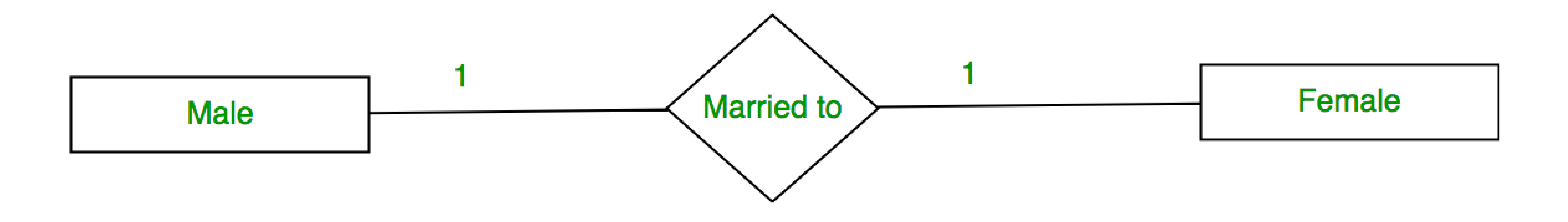


Cardinality:

- The **number of times an entity of an entity set participates in a relationship** set is known as cardinality. Cardinality can be of different types:

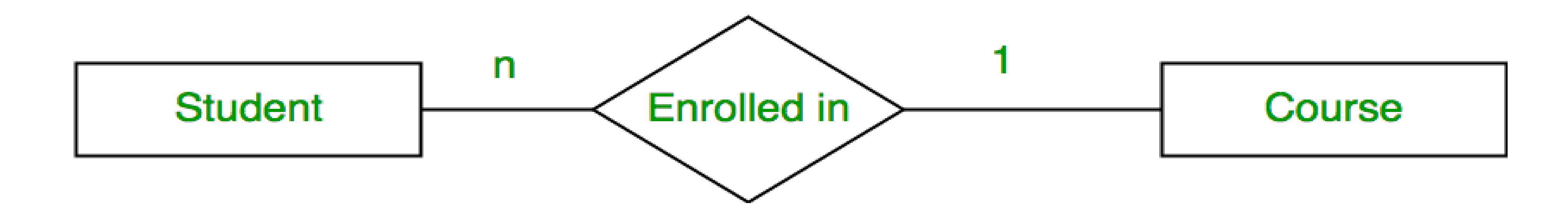
1. One-to-one

- When each entity in each entity set can take part **only once in the relationship**, the cardinality is one-to-one. Let us assume that a male can marry one female and a female can marry one male. So the relationship will be one-to-one.
- the total number of tables that can be used in this is **2**.



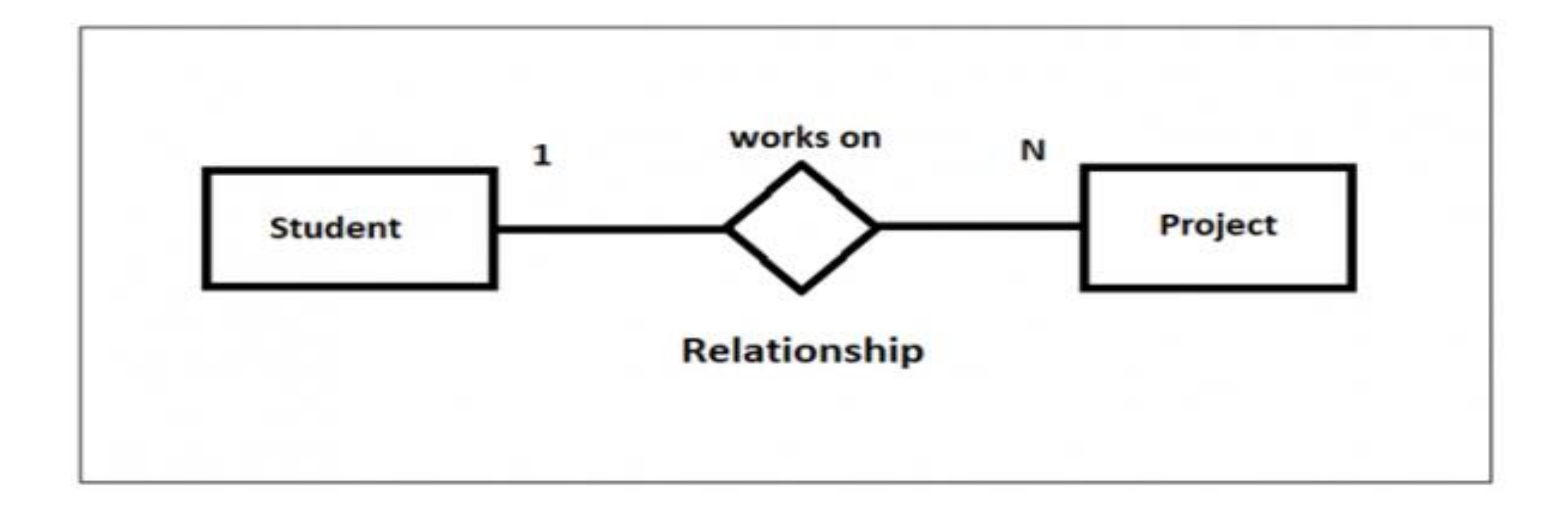
2. Many to one

- When entities in one entity set **can take part only once in the relationship set and entities in other entity sets can take part more than once in the relationship set,** cardinality is many to one. Let us assume that a student can take only one course but one course can be taken by many students. So the cardinality will be n to 1. It means that for one course there can be n students but for one student, there will be only one course.
- The total number of tables that can be used in this is **3**.



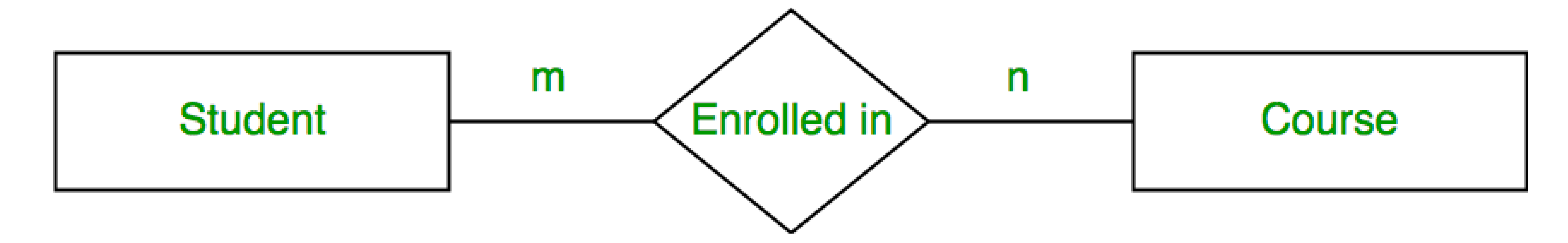
3. One to Many

- Eg: One student can work on many projects. So it is one to many relation.



4. Many to many

- When entities in all entity sets can **take part more than once in the relationship** cardinality is many to many. Let us assume that a student can take more than one course and one course can be taken by many students. So the relationship will be many to many.
- the total number of tables that can be used in this is **3**.

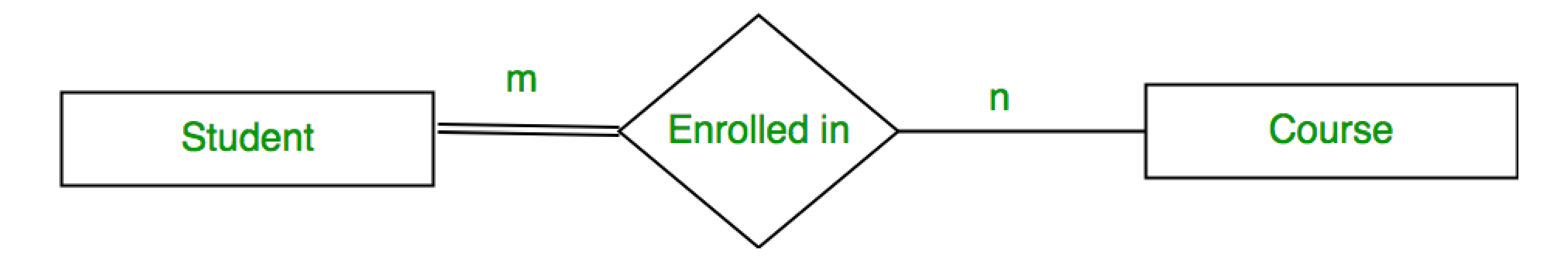


Participation Constraint:

- Participation Constraint is applied to the entity participating in the relationship set.

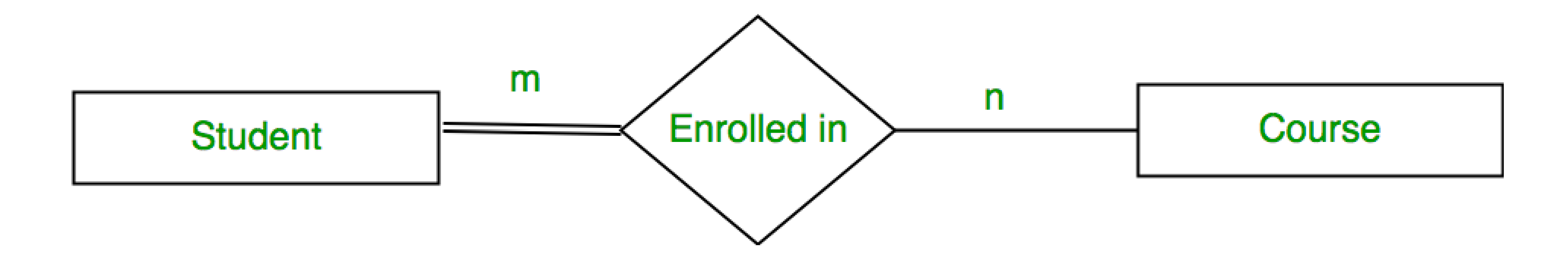
1. Total Participation

- Each entity in the entity set **must participate** in the relationship. If each student must enroll in a course, the participation of students will be total. Total participation is shown by a double line in the ER diagram.



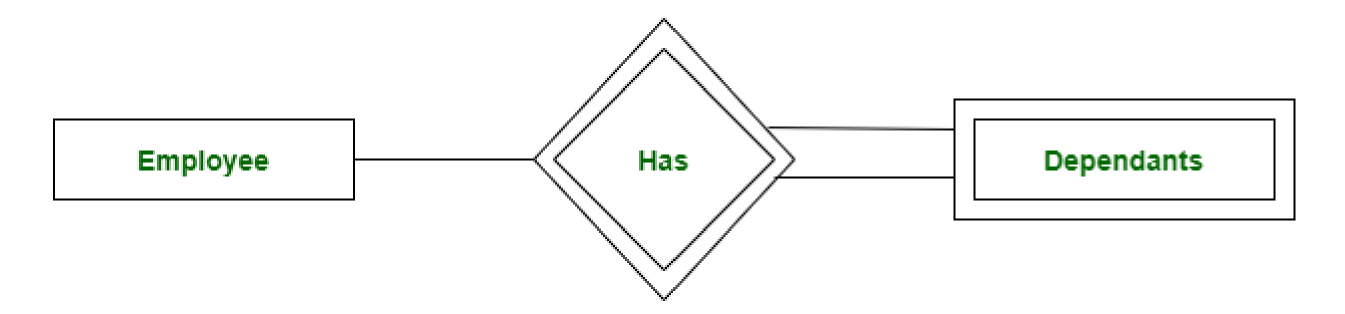
2. Partial Participation

- The entity in the entity set **may or may NOT participate** in the relationship. If some courses are not enrolled by any of the students, the participation of the course will be partial.
- The diagram depicts the 'Enrolled in' relationship set with Student Entity set having total participation and Course Entity set having partial participation.



Weak Entity Type and Identifying Relationship:

- A company may store the information of dependents (Parents, Children, Spouse) of an Employee. But the dependents don't have existed without the employee. So Dependent will be a weak entity type and Employee will be Identifying Entity type for Dependent.
- A weak entity type is represented by a double rectangle. The participation of weak entity types is always total. The relationship between the weak entity type and its identifying strong entity type is called identifying relationship and it is represented by a double diamond.



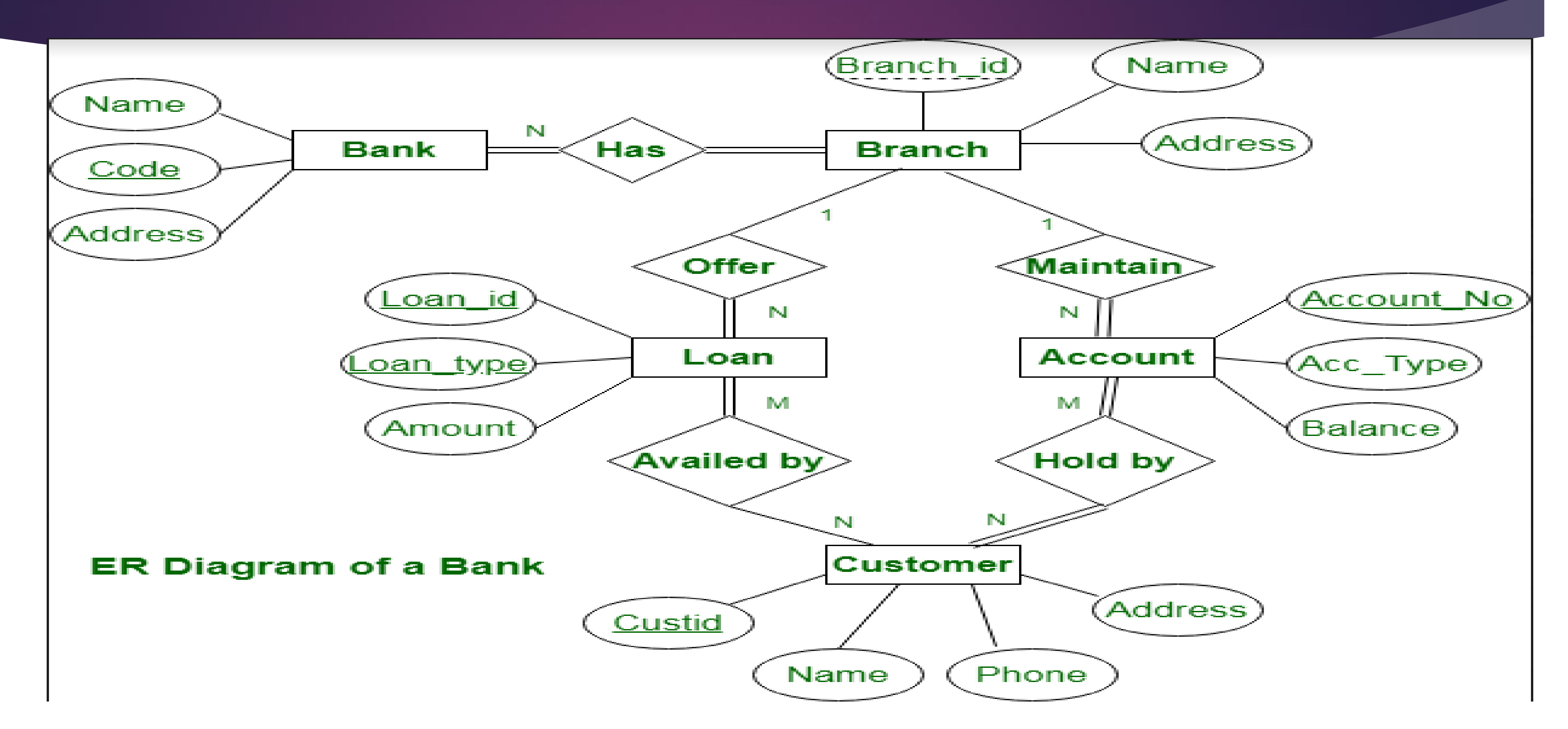
ER diagram of Bank Management System

- ER diagram is known as Entity-Relationship diagram. It is used to analyze to structure of the Database. It shows relationships between entities and their attributes. An ER model provides a means of communication.

ER diagram of Bank has the following description:

- Bank have Customer.
- Banks are identified by a name, code, address of main office.
- Banks have branches.
- Branches are identified by a branch_no., branch_name, address.
- Customers are identified by name, cust-id, phone number, address.
- Customer can have one or more accounts.
- Accounts are identified by account_no., acc_type, balance.
- Customer can avail loans.
- Loans are identified by loan_id, loan_type and amount.
- Account and loans are related to bank's branch.

ER Diagram : Bank Management System



Entities and their **Attributes** are :

- **Bank Entity :** Attributes of Bank Entity are Bank Name, Code and Address.
  Code is Primary Key for Bank Entity.
- **Customer Entity :** Attributes of Customer Entity are Customer_id, Name, Phone Number and Address.
  Customer_id is Primary Key for Customer Entity.
- **Branch Entity :** Attributes of Branch Entity are Branch_id, Name and Address.
  Branch_id is Primary Key for Branch Entity.
- **Account Entity :** Attributes of Account Entity are Account_number, Account_Type and Balance.
  Account_number is Primary Key for Account Entity.
- **Loan Entity :** Attributes of Loan Entity are Loan_id, Loan_Type and Amount.
  Loan_id is Primary Key for Loan Entity.

Relationships are :

- **Bank has Branches => 1 : N**
  One Bank can have many Branches but one Branch can not belong to many Banks, so the relationship between Bank and Branch is one to many relationship.

- **Branch maintain Accounts => 1 : N**
  One Branch can have many Accounts but one Account can not belong to many Branches, so the relationship between Branch and Account is one to many relationship.

- **Branch offer Loans => 1 : N**
  One Branch can have many Loans but one Loan can not belong to many Branches, so the relationship between Branch and Loan is one to many relationship.

- **Account held by Customers => M : N**
  One Customer can have more than one Accounts and also One Account can be held by one or more Customers, so the relationship between Account and Customers is many to many relationship.

- **Loan availed by Customer => M : N**
  (Assume loan can be jointly held by many Customers).
  One Customer can have more than one Loans and also One Loan can be availed by one or more Customers, so the relationship between Loan and Customers is many to many relationship.

THANK YOU

